

# LOAD'N'RUN

RIVISTA DI PROGRAMMI SU CASSETTA PER IL TUO COMPUTER



**16**

**GIOCHI  
& UTILITY per**

**SPECTRUM**



GENERATOR - SFIDA NBA - ZAP LAP - SEXY BLACK JACK - TENNIS L/M - SLOT MACHINE  
ESERCIZI VERBI INGLESI - TOTOTOOLKIT - 3003 - CATERINA - RIFORMA SANITARIA  
BARMAN - PUNCHBALL - L'OCCHIO DEL DIAVOLO - PER ELISA - DIAISTO

# ECCEZIONALE NOVITÀ

IN TUTTE LE EDICOLE

## TUTTO QUEL CHE SERVE PER ENTRARE NEL FAVOLOSO MONDO DELLA COMPUTER-COMUNICAZIONE VIA TELEFONO

Hai un computer e un telefono? Questa rivista è fatta apposta per te! Tutto l'hardware e il software (su cassetta!) per comunicare via computer e telefono in Italia e in tutto il mondo. Potrai metterti in contatto con altri computer, trasmettere e ricevere posta, catturare programmi, trovare i mille amici (gli hackers!) del Modem Club International con cui scambiare programmi e informazioni... più una banca dati diretta a disposizione 24 ore... più altro ancora...

la prima rivista per computer via telefono

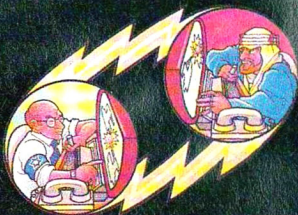
# MODEM

## COMPUTER MAGAZINE

N. 1 GENNAIO 1986

Sped. in abb. post. gr. III - L. 9.000

PIÙ  
I NUMERI  
DI TELEFONO  
PER PORSI  
IN  
CONTATTO



CON IL  
FASCICOLO,  
GRATIS  
IL SOFTWARE  
SU  
CASSETTA

**COSTRUISCI IL TUO MODEM!**

**MODEM CLUB**

**DATA HACKERS NEWS**

**MICROMARKET LETTORI**

**LE INTERFACCE UTILI**

**MODEM MERCATO**

da casa  
**BANCA  
DATI**

con il software  
**IN CASSETTA  
OMAGGIO!**

Questo fascicolo può anche essere richiesto  
con vaglia postale di Lire 10.000 ad Arcadia, c.so Vittorio Emanuele 15, 20122 Milano



L'immagine di copertina è stata realizzata da Milko Mrsek (Brescia)

N. 24 - FEBBRAIO 1986

Direttore  
Mario Magrone

Redattore Capo  
Sira Rocchi

Direzione Tecnica  
Nadia Marini

Stampa  
Garzanti Editore S.p.A.  
Cernusco S/N (MI)

Distribuzione  
SO.DI.P. Angelo Patuzzi srl  
Via Zuretti 25, Milano

Amministrazione, Redazione, Pubblicità: Arcadia s.r.l., C.so Vittorio Emanuele 15, 20122 Milano. Una copia lire 9.000, arretrato lire 10.000. Fotocomposizione: Composit. Selezione colori e fotolito: Eurofoto. Stampa: Garzanti Editore S.p.A. Milano. Distribuzione: SO.DI.P. Angelo Patuzzi srl, Via Zuretti 25, Milano. Load 'N' Run è un periodico mensile registrato presso il Tribunale di Milano al numero 580 in data 24 dic. 83. Resp. Mario Magrone. Spedizione in abbonamento postale Gr. III/70. Pubblicità inferiore al 70%. Tutti i diritti sono riservati per tutti i paesi. Manoscritti, disegni, fotografie e programmi inviati non si restituiscono anche se non pubblicati. Rights reserved everywhere.

# LOAD'N'RUN

RIVISTA SU CASSETTA DI PROGRAMMI PER COMPUTER

## SOMMARIO

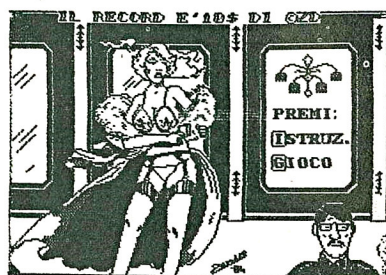
- |                          |                        |
|--------------------------|------------------------|
| ■ GENERATOR              | ■ 3003                 |
| ■ SFIDA NBA              | ■ CATERINA             |
| ■ ZAP LAP                | ■ RIFORMA SANITARIA    |
| ■ SEXY BLACK JACK        | ■ BARMAN               |
| ■ TENNIS L/M             | ■ PUNCHBALL            |
| ■ SLOT MACHINE           | ■ L'OCCHIO DEL DIAVOLO |
| ■ ESERCIZI VERBI INGLESI | ■ PER ELISA            |
| ■ TOTOTOOLKIT            | ■ DIAISTO              |

Per caricare i programmi riavvolgete il nastro e date LOAD", per uscire dai giochi togliete l'alimentazione e digitate nuovamente LOAD". Le cassette utilizzate per registrare i programmi sono tutte di ottima qualità: assicuriamo comunque (rivolgetevi direttamente alla redazione *esclusivamente scrivendo*) la sostituzione di eventuali cassette difettose. Lo ZX Spectrum è un computer prodotto dalla Sinclair Research Ltd. (UK). È distribuito in Italia dalla GBC Italiana SpA.



# SEXY BLACK JACK (48K)

di D. Zambrini



Questa volta dovrai giocare al famoso gioco delle carte «Black Jack» contro il computer, ma in una versione piuttosto piccante...

Devi riuscire a fare il punteggio di 21 (cifra ricavata dalla somma delle carte) ricordandosi che le figure valgono 10, mentre l'asso può valere sia 1 che 11. Se il punteggio è superiore a 21 perdi la puntata e la ragazza si riveste. In caso di pareggio è il Banco (lo Spectrum) che vince.

Tasti:

G = inizio gioco

6 = diminuisce la puntata da giocare

7 = aumenta la puntata da giocare

0 = accetta la puntata selezionata

S = richiesta di un'altra carta

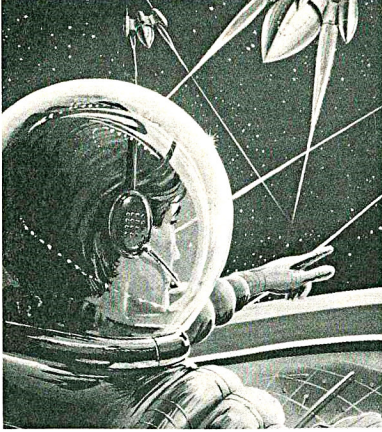
N = passa la mano al computer

Il listato è adatto a capire cosa sono le funzioni.

Per funzione s'intende un particolare comando che, nella maggior parte dei casi, necessita di un operando sul quale calcolare il risultato. Lo Spectrum offre ben 31 funzioni già implementate nel sistema con un nome proprio (es. SIN, COS, SQR, VAL, ecc.) più la possibilità di definirne altre con il comando DEF FN e quindi di poterle utilizzare (FN).

Anche in questo programma sono state usate più volte diverse

```
770)LET x=USR "3"
780 FOR k=0 TO 31: POKE x+k,0:
NEXT k
790 LET s=8*CODE b$(c)+15360: I
b$(c)="7" THEN LET s=USR "e"
800 FOR j=0 TO 7: LET v=PEEK (s
+j): LET c=0: LET d=256
810 FOR k=8 TO 1 STEP -1: LET d
=d/2: IF v)=d THEN LET c=c+3*d+d
: LET v=v-d
820 NEXT k
830 LET h=INT (c/256): LET i=c-
256+h
840 POKE x,h: POKE x+1,h: POKE
x+8,i: POKE x+9,i
850 LET x=x+2+8*(j=3)
860 NEXT j
870 INK 7: PRINT AT p-3,q:"
": FOR k=1 TO 5: PRINT AT p-3+
k,q:"": NEXT k: PRINT AT p
+1,q:"": REEP .04,-30: INK
0
880 LET ma=1+INT (4*RND): INK 2
*(ma<3): PRINT AT p-2,q+3: PAPER
7: CHR$ (148+ma)
890 PRINT AT p,q+2: PAPER 7:"47
": AT p+1,q+2:"1"
900 RETURN
910 LET pu=INT (d0/2): IF pu=0
THEN LET pu=1
920 PRINT AT 19,22:pu:"
930 IF INKEY$="7" OR IN 31=8 TH
EN LET pu=pu+1: REEP .01,10: IF
pu>=32 THEN LET pu=0
940 IF INKEY$="6" OR IN 31=9 TH
EN LET pu=pu-1: REEP .01,0: IF p
u<=1 THEN LET pu=1
950 IF INKEY$="0" OR IN 31=16 T
HEN PRINT AT 19,22: FLASH 1:pu:
REEP .04,q+3: REEP .04,q+6: REEP
.04,q+3: REEP .2,q: LET s4="v"
beeme,: GO TO 250
960 GO TO 890
970 PARSE FN X(2)
```



# GENERATOR (48K)



**A**luni alieni senza scrupoli e con un alto senso dell'umorismo, decidono di eliminare per sempre dall'universo tutte le forme di vita intelligente. Spediscono dunque verso la Terra una gigantesca astronave per distruggerci.

Il tuo compito è di penetrare nell'astronave, evitare tutte le creature ed i trabocchetti, sabotare

il generatore centrale e... non farti neanche un graffio perché la mutua non caccia una lira.

Tasti:

7 = Destra

6 = Sinistra

H = Pausa

S = Riparti

P = Fine

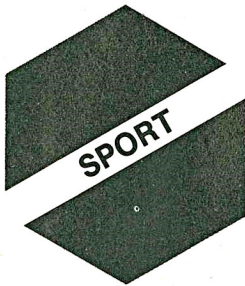
Joystick Kempston.

funzioni, vediamo per esempio alla linea 790 l'uso della funzione CODE. Quest'ultima ha come operando una stringa e dà come risultato il codice ASCII del primo carattere della stringa o 0 se la stringa è vuota. A pagina 135 del manuale inglese vi è la tabella dei caratteri dello Spectrum con il loro corrispondente numero. Il carattere 'A' ha come codice 65 mentre lo spazio il codice 32. Utilizzando la funzione CODE non sarà più necessario consultare questa tabella. Infatti se volessimo sapere il codice del carattere '+' è sufficiente digitare il comando diretto PRINT CODE "+" e il risultato sarà 43. Ma la funzione CODE si rivela molto utile nel riconoscimento di quei caratteri non stampabili o per lo meno non visualizzabili. Supponiamo, per esempio, di dover eseguire una certa parte di programma solo se viene premuto il tasto ENTER, il programmino di gestione sarà:

```
1000 LET A$=INKEY$
1010 IF CODE A$<>13 THEN
GOTO 1000
```

1020 resto del programma: è stato premuto ENTER

Il codice ASCII del carattere ENTER è appunto 13 e quindi il programma passa alla linea 1020 solo quando viene premuto ENTER.



# SFIDA NBA (48K)

**E**cce una fantastica sfida nell'NBA Americano.

Puoi decidere di essere DR.J che sfida Larry Bird o viceversa.

Tramite le opzioni puoi scegliere se giocare per un certo periodo di tempo o se continuare fino ad

un altro giocatore anziché contro il computer.

Tasti:

Larry Bird

DR.J

L

Alto

Q

M

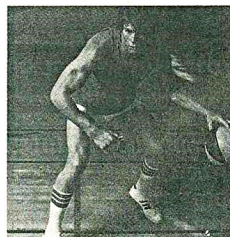
Basso

A

M

Destra

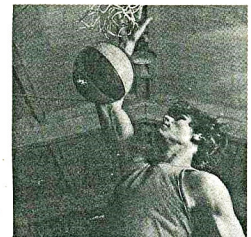
X



un certo punteggio.

Puoi anche decretare che la palla, dopo il canestro, vada al «vincente» o al «perdente».

Ci sono quattro livelli di difficoltà e puoi giocare anche contro



N

Sinistra

Z

B

Tiro/Salto

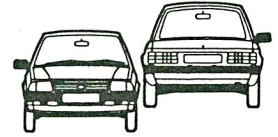
C

Nel menù delle opzioni usa i tasti di DR.J; la C cambia da destra a sinistra la posizione del cursore (lampeggio) e sceglie l'opzione.



# ZAP LAP (48K)

di A. Broggi



**F**antastico! Un sofisticato simulatore di guida auto che ti permetterà di provare la tua abilità come pilota. La simulazione prende in considerazione il motore FIRE montato sulla Fiat Uno 45. Al termine della simulazione, il computer elaborerà un referto che riporterà lo stato finale della vettura, gli errori più frequenti del pilota ed alcuni consigli per una guida migliore. La prova si conclude al compimento del centesimo chilometro o in seguito a guasto alla macchina. Contribuiscono a peggiorare lo stato della vettura le seguenti inosservanze: transitare sulle corsie di emergenza; circolare a bassa velocità con una marcia alta; l'eccessivo uso della frizione e, naturalmente, gli incidenti. Lo stato della vettura è indicato da una spia rettangolare sulla destra dei contagiri.

- Tasti:  
 X = acceleratore  
 Z = freno  
 Caps shift = frizione  
 R = sinistra  
 T = destra  
 1, 2, 3, 4, 5 = cinque marce  
 Symbol shift = motorino d'avviamento  
 0 = termine anticipato della prova

```

200>REM AAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAA ????A?AAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
205 POKE 23693,64: POKE 23624,6
1: OUT 254,0: CLS
2: 210 LET ATTA=99996: LET SCOM=999
3: LET LDIA=9999
4: 220 PAUSE 100
5: 230 LET HL=55405: GO SUB SCOM
6: 240 LET A=70: LET HL=22528: LET
BC=256: GO SUB ATTA
7: 250 PAUSE 150
8: 260 LET A=70: LET HL=22784: LET
BC=256: GO SUB ATTA
9: 270 PAUSE 150
10: 280 CLS
11: 290 PAUSE 100
12: 300 FOR F=5 TO 9: PRINT AT F,9:
INK 4;" "; BRIGHT 0; INK 7;
NEXT F
13: 310 FOR F=135 TO 95 STEP -1: PL
OT INK 8;72,F: DRAW INK 8;119,0:
NEXT F
14: 320 FOR F=5 TO 9: PRINT INK 0;9
F,9; PAPER 4;" "; BRIGHT 3
; PAPER 7;" "; BRIGHT 1; PAP
ER 2;" "; NEXT F
15: 330 PAUSE 100
16: 340 LET HL=53025: GO SUB SCOM
17: 350 PAUSE 170
18: 360 FOR F=95 TO 135: PLOT INK 8
OVER 1;72,F: DRAW INK 8; OVER
1;119,0: NEXT F
19: 370 PAUSE 100: FOR F=5 TO 9: PR
INT OVER 1;AT F,9; PAPER 0; INK
6; OVER 1;" "": NEX
T F

```



# CATERINA (16K)



**C**aterina ha deciso di andare nel bosco a raccogliere dei funghi. Ella non sa che ci sono dei terribili lupi e sassi stregati che la possono uccidere.

Aiuta Caterina e... in bocca al lupo!

Tasti:  
 da Caps a Space = destra/sinistra  
 da A a Enter = giù

H = sospensione temporanea  
S = fine sospensione.

Per il programma: sono diverse le istruzioni interessanti.

Solitamente per definire i colori della carta e dell'inchiostro ricorriamo alle istruzioni PAPER ed INK. Così per la luminosità e per il lampeggio usiamo BRIGHT e FLASH. Invece è possibile ottenere l'effetto delle quattro istruzioni con una semplice POKE. Ovviamente andiamo a modificare una variabile di sistema, precisamente ATTR-P il cui indirizzo è 23693. ATTR-P contiene i valori degli attributi permanenti di FLASH, BRIGHT, PAPER ed INK. Il comando CLS, per esempio, cancella i contenuti della RAM video tenendo conto dei colori definiti in ATTR-P. Vediamo il significato di ogni bit di questa variabile di sistema:

bit n. 7 = FLASH; se 1 è FLASH 1 se 0 è FLASH 0

bit n. 6 = BRIGHT; se 1 è BRIGHT 1 se 0 è BRIGHT 0

bit n. 5, 4 e 3 = PAPER; contengono il valore binario del colore della carta. Per esempio, per PAPER 4 i bit 5, 4 e 3 contengono il valore 100.

bit n. 2, 1 e 0 = INK; contengono il valore binario del colore dell'inchiostro. Per esempio, per INK3 i bit 2, 1 e 0 contengono 011.

In questo programma, alla linea 206, l'istruzione POKE 23693,64 setta FLASH 0, BRIGHT 1, PAPER 0 ed INK 0. Infatti la rappresentazione binaria di 64 è 01000000 e quindi solo il bit n. 6 è settato, cioè quello relativo a BRIGHT. Invece nei tre bit di PAPER, così come in quelli di INK, vi è il valore 000 corrispondente al colore nero.



da Q a P = su

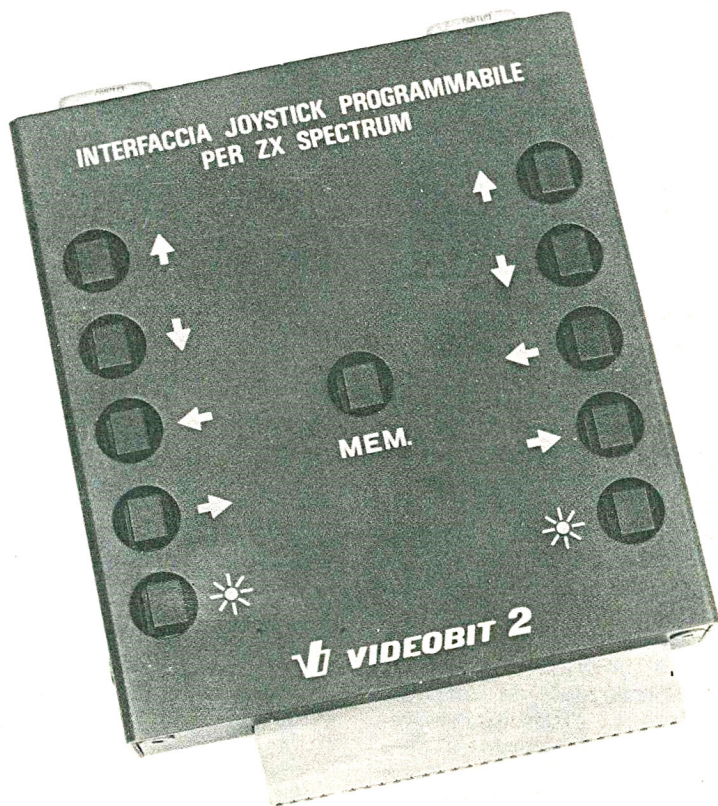
3 = pausa

4 = riparte dopo la pausa

0 = restart.

## UN JOYSTICK PER TE!

*Giocare premendo tasti non è bello, né per il giocatore né per la povera tastiera dello Spectrum: bisogna equipaggiarsi di un buon joystick e della relativa interfaccia. Di joystick se ne trovano più o meno pratici, più o meno costosi. Di interfacce poi ce ne sono di normali e di super. La Videobit (02/390516) ha deciso di regalarne una veramente super a quello tra voi che, inviando un programma, si distinguerà per... originalità. Insomma, questa volta non basta essere bravi, fare un bel programma ricco di grafica e suono*



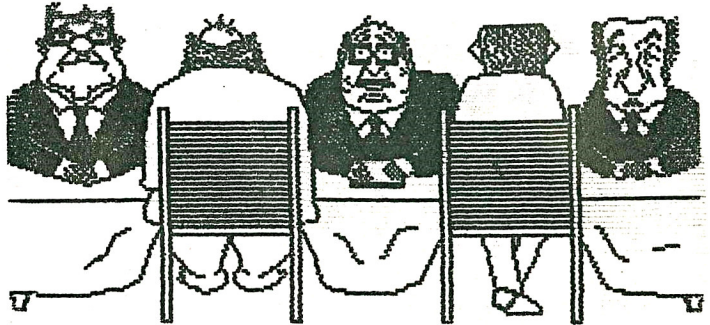
e perfettamente funzionante. Questa volta viene premiata l'idea. Scatenate la fantasia dunque, se volete che il videojoystick sia vostro: la redazione sceglierà insindacabilmente e pubblicherà il vostro lavoro con il vostro nome. Due parole ora su questa sofisticatissima interfaccia. Collegamento diretto e simultaneo di due joystick; possibilità di programmare i tasti funzione in tutte le direzioni, veramente tutte! Non come certe interfacce con le quali gli spostamenti avvengono solo verso il basso e l'alto; con questa si programmano spostamenti in ogni direzione, anche lungo le diagonali. Grazie a questa prerogativa, si dispone di un joystick che manovra le operazioni dello schermo con una facilità ed una velocità incredibili. La programmazione è rapidissima e passare da un gioco ad un altro in cui cambiano i tasti funzione è cosa di attimi.



# RIFORMA SANITARIA (48K)

di G.P. Antonello

Un gioco davvero nuovo e affascinante. Siedi anche tu in Parlamento nei panni del Ministro della Sanità: devi riuscire, con un certo numero di tentativi e in un dato tempo, a far approvare dal Parlamento stesso l'ennesima Riforma Sanitaria. La Riforma è una frase divisa in sette parti (soggetto, verbo, complementi vari). Si hanno a disposizione 10 soggetti (da 0 a 9), 10 verbi, 10 complemento oggetto e così via. Ogni frase è quindi riconducibile ad un numero di sette cifre in cui la prima rappresenta il soggetto, la seconda il verbo, la terza il complemento oggetto ecc. Si dovrà affrontare per primo il Consiglio dei Ministri a cui seguiranno i vari passaggi obbligati di una nuova legge: vertici politici, incontri con le forze sociali, camera, senato... In ogni frase sarà possibile presentare una nuova proposta di legge (introducendo il codice numerico) o accogliere una controproposta presentate da ministri, partiti, ecc. La proposta di legge verrà accolta e approvata dal Senato quando il codice introdotto coinciderà con quello della legge definitiva (generato inizialmente dal computer).



```

0360 DATA "4" "14" "4" "14" "2"
"9" "5" 333 "10" "15" "12" "5" 333
"3" "10" "16" "9" "2" "7" "4" "10"
"4" "5" 333 "2" "9" "15" "10" "2" "10"
"5" "5" 333 "2" "5" "14" "4" "15" "4"
0370 DATA "0" "4" "14" "5" 333
"3" "14" "16" "15" "2" "14" "4" "10"
"14" "5" 333 "17" "16" "15" "4" "10"
"14" "8" "14" "8" "16" "15" "4" "10"
"3" "17" "4" "10"
0450 DEF FN 0()=INT ((65536*PEEK
23674+255*PEEK 23673+PEEK 23672
)/50): POKE 23674,0: POKE 23673,
0: POKE 23672,0: RETURN
0460 LET T=INT (FN 0()/60): IF t
<=tp THEN RETURN
0470 LET tp=t: PRINT AT NOT PI,5
: INK 9: INVERSE 1: FLASH 1 AND
tp>=tem-2;tp: BEEP .05,40
0480 IF tp>=tem THEN GO TO 6000
0490 RETURN
0700 LET k=SGN PI+INT (RND*7): P
RINT AT SGN PI,17;Z$(15 TO 17)

```

Vi sono 4 livelli di difficoltà:  
1 = 20 tentativi in 30 mesi (minuti)  
2 = 16 tentativi in 25 mesi (minuti)  
3 = 13 tentativi in 17 mesi (minuti)  
4 = 10 tentativi in 12 mesi (minuti)

L'inserimento del codice viene fatto introducendo i 7 numeri (da 0 a 9) formulanti la proposta di Legge. Fatto questo il Presidente del Consiglio (o il tabellone delle

sale parlamentari) comunicherà quante parti della riforma sono approvabili. Ecco il prospetto:  
1 - Disastrosa; 2 - Inconsistente; 3 - Lacunosa; 4 - Utopistica; 5 - Costruttiva; 6 - Interessante; 7 - Accolta. Per esempio, se la risposta è «Disastrosa» vuol dire che solo il soggetto è esatto; se invece la risposta è «Lacunosa», vuol dire che oltre al soggetto anche due parti di frase (numeri) sono esatte (numero giusto al posto giusto), si tratta di scoprire quali.

Premendo il tasto «A» si possono accogliere controproposte presentate da altri ministri, partiti, forze sociali ecc. In tal caso si perderà un tentativo, ma chissà la fortuna potrebbe essere benigna.

Al termine del gioco verrà visualizzata la legge definitiva.

Il programma è composto da 4 blocchi:  
1) programma caricatore  
2) screen  
3) linguaggio macchina  
4) Basic



Gli ultimi tre blocchi sono stati salvati senza header.

L'header è l'insieme delle informazioni che precedono ogni blocco di dati registrato, sia esso un programma Basic o codici di linguaggio macchina.

L'header contiene il tipo del blocco (Basic o codici), il nome con cui è stato registrato, la sua lunghezza e così via. L'header è lungo 17 bytes, ognuno di essi contiene una particolare informazione. Vediamo separatamente come

byte n. 1 = (sta ad indicare blocco codici)

dal byte n. 2 al byte n. 11 = nome del blocco

byte n. 12-13 = lunghezza blocco

byte n. 14-15 = indirizzo d'inizio del blocco

byte n. 16-17 = non usati.

Il terzo blocco contiene diverse routine in linguaggio macchina tra le quali quella per il caricamento del Basic. Quest'ultima inizia all'indirizzo 64879. Per caricare il blocco del linguaggio macchina



# L'OCCHIO DEL DIAVOLO

(48K)

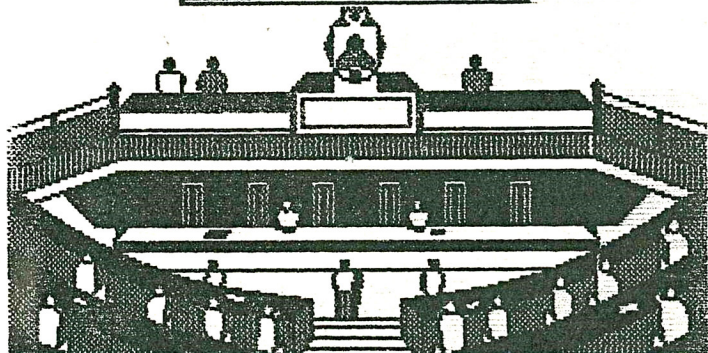
di A. Savio

```

3950 FOR n=VAL "6" TO VAL "12":
POKE dr,n: FOR m=7 TO 2 STEP -1:
POKE al,m: RANDOMIZE USR ss: BE
EP .005,30+15-m: NEXT m: NEXT n:
RETURN
3001 LET xx=(256-8*xs#LEN P$)/2
9010 LET i=23306: POKE i,xx: POK
E i+1,yy: POKE i+2,xs: POKE i+3,
ys: POKE i+4,8: LET i=i+4: LET w
=LEN P$
9020 FOR f=1 TO w: POKE i+f,CODE
P$(f): NEXT f
9030 POKE i+w+1,255: LET w=USR 6
3915: RETURN

```

|            |     |
|------------|-----|
| FAVOREVOLI | 266 |
| CONTRARI   | 52  |
| ASTENUTI   | 17  |
| APPROVATA  |     |



è strutturato l'header del Basic e quello dei codici.

Header Basic:

byte n. 1 = 00 (sta ad indicare Basic)

dal byte n. 2 al byte n. 11 = nome del programma

byte n. 12-13 = lunghezza programma con le variabili

byte n. 14-15 = linea di autostart

byte n. 16-17 = lunghezza programma senza variabili

Header codici (CODE, CREEN\$, DATA()):

occorre eseguire il seguente programmino Assembler:

ORG 49060

LD DE, 16470

LD IX, 49066

LD A, 255

SCF

CALL 0556H

RET

Utilizzando poi la routine di caricamento Basic si osservino le linee 9001-9030. Questa parte di programma consente anche di stampare scritte giganti.

**E**ra una notte buia e tempestosa... e tu veleggiavi impavido con una goletta alla volta dell'Isola Senza Nome. Un vecchio marinaio ti ha detto che l'isola custodisce un antico segreto: un enorme, preziosissimo rubino grande quanto un uovo di struzzo e chiamato «l'occhio del diavolo» giace nascosto nell'isola. La leggenda dice (e non sbaglia!) che a guardia della preziosa pietra c'è un idolo di alabastro dai poteri misteriosi. Tu approdi sull'isola e ti trovi su una bellissima spiaggia, dalla quale parti alla ricerca del tesoro. Premi ENTER per vedere il luogo in cui ti trovi o per tornare alla descrizione. Inserisci sempre verbo ed oggetto tralasciando preposizioni ed articoli (esempio: PRENDO BORRACCIA, REGALO FICHES etc.). Durante il giorno fa un caldo torrido, quindi fai attenzione alla tua riserva d'acqua! Per muoverti basta tu inserisca: N per nord; S per sud; E per est; O per ovest.

Un programma Basic viene salvato su nastro con il comando

SAVE «nome programma» oppure nella forma con autostart alla linea n SAVE «nome programma» LINE n. Non tutti sanno, invece, che un programma Basic può essere salvato come blocco CODE. Prendiamo in considerazione questa avventura il cui programma Basic è stato salvato come blocco CODE. La linea interessata è la numero 9980. Osserviamo che prima di eseguire l'istruzione SAVE viene calcolata la variabile A (valore casuale minore di 256) e la variabile S. Quest'ultima contiene il valore della variabile di sistema STKEND (indirizzo 23653/4) contenente l'indirizzo dove inizia la memoria libera, cioè non ancora utilizzata dal Basic.

È facile ora comprendere che salvando l'intera area di memoria a partire dall'inizio delle variabili di sistema (indirizzo 23552) fino all'indirizzo indicato a STKEND si salva l'intero programma Basic con i suoi puntatori e stacks cosicché quando il blocco CODE viene ricaricato la sua esecuzione riparte dall'istruzione successiva al comando SAVE. In questo programma l'istruzione successiva a SAVE è GOTO 9490, quindi una volta caricato il blocco «Basic» il programma passerà all'esecuzione dell'istruzione GOTO 9490. Non vi preoccupate se alla linea 9980 il programma viene salvato come SAVE «OcchioCODE 3»CODE

```

9975)LET MAG=0: LET PRE=1: LET P
#="1111": LET LI=10: LET X=3: LE
T Y=8: RETURN
9980 PAPER 1: INK 1: CLS : LET S
=PEEK 23653+256*PEEK 23654: LET
A=INT (RAND*256): SAVE "OcchioCOD
E 3"CODE 23552-A,S-23500: GO TO
9490
9985 FOR N=64500 TO 64500+73: PR
INT N,PEEK N: NEXT N
9988 LET ACC=12: FOR N=1 TO 12:
LET O(N)=2: NEXT N
9989 GO SUB 7200: GO SUB 7300: G
O TO X+1000+Y+100
9990 CLEAR
9991 POKESIN e LET SUB 7777)NEXT 7777
9992 COS a) POKE 7777)INKEY$)FOR ??
7777)INKEY$)a BORDER 7777)VAL KLEN
7776)VAL KSIN 7777)VAL$ r)LEN ?
9993 X??)VAL$ r)MSIN ?? CLS ??COS t)LE

```



```

60 IF G$=G$ THEN LET WW=7: GO
SUB 0: PRINT PAPER 7:"BENE! ORA
HAI LA PANCIA PIENA !!": GO TO 6
000
65 LET WW=7: GO SUB 0: PRINT P
APER (W);X$: GO TO 6000
90 POKE PK1,0: POKE PK2,60: RE
TURN
95 POKE PK1,STR1: POKE PK2,STR
2: GO SUB 9500+X*10+Y: GO SUB 90
RETURN.

```

23552-A,S-23500, la variabile A, essendo casuale, salva ogni volta un blocco con start differente, potrebbe essere eliminata, l'importante è che lo start sia 23552. Allo stesso modo la lunghezza del codice da salvare potrebbe essere minore: S-23552 e non S-23500.

tante è che lo start sia 23552. Allo stesso modo la lunghezza del codice da salvare potrebbe essere minore: S-23552 e non S-23500.



**3003 (48K)**



Corre l'anno 3003 e tu sei stato scelto quale protagonista di una tremenda missione: recuperare, nel labirinto dell'automation, tutti gli oggetti di produzione umana che trovi. Nessuno è riuscito a tornare da quella impresa, provaci tu!

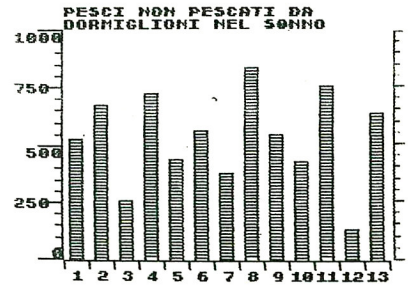
Tasti:  
M = Destra  
N = Sinistra  
A = Su  
Z = Giù  
da Caps a Space (tranne N e M) = fuoco.



# DIAISTO

## (48K)

di C. Molinari



Permette di tracciare Diagrammi e Istogrammi fissando il valore massimo della variabile da rappresentare (asse verticale y) e il numero di valori che si vogliono assegnare a questa variabile. Per esempio, fissando il valore max a 9000 e assegnando ad n il valore 12, si potrà rappresentare graficamente (dia o isto) l'andamento delle vendite mensili (ovviamente in migliaia di copie!) di Load'n'Run nel mondo. Potrete anche far calcolare e rappresentare in istogrammi la distribuzione percentuale mensile delle vendite della rivista.

Numerosissime opzioni consentono di cambiare la rappresentazione grafica utilizzando campiture diverse, per colore e retinatura, di inserire scritte, di utilizzare griglie ed evidenziatori, di tracciare più di un diagramma o istogramma in uno stesso grafico, di richiamare i valori inseriti, di stampare su carta in modi diversi, di salvare su nastro i grafici, ecc. Il programma, nella parte iniziale, fornisce tutte le istruzioni necessarie alla scelta del tipo di grafico e all'inserimento dei valori. Per il completamento dei grafici un menù indica le diverse possibilità. Le istruzioni che seguono - e che possono essere sempre richiamate nel corso del programma - spiegano il significato dei simboli indicati nel menù. Con un po' di pratica potrete ottenere ottimi grafici e, soprattutto, data la grande flessibilità del programma, potrete adattarlo a tutte le esigenze (numero di valori e tipo di variabile) che richiedano la rappresentazione grafica di una o più variabili.

### ISTRUZIONI

I Richiama le istruzioni.

- T Richiama la tabella valori.
- M Solo Dia: Elimina i diagrammi tracciati, uno alla volta.
- B Pulisce il grafico: azzerà dia e isto mantenendo gli assi.
- A Traccia altri dia o isto nello stesso grafico.
- G Traccia assi di riferimento.

### CAMPITURE DIAGRAMMI

- 1 Righe verticali a densità variabile (dens. 1 = pieno).
- 2 Tratteggio a densità variabile.
- 3 Righe verticali a partire dall'Alto del grafico.
- 4 Tratteggio a partire dall'Alto del grafico.

```

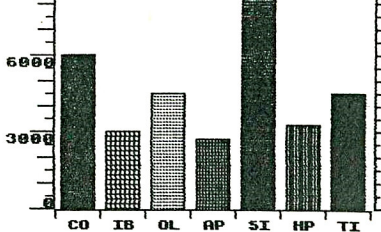
2055 FOR Y=10 TO 150 STEP 5
2060 IF POINT (X,Y)=1 THEN GO TO 2085
2065
2065 IF POINT (X,Y-1)=1 THEN GO TO 2085
2070 IF POINT (X,Y-2)=1 THEN GO TO 2085
2075
2075 PLOT OVER 1; INK C1;X,Y
2080 NEXT Y
2085 NEXT X
2090 GO TO VAL "9000"
2100 INPUT "COLORE CAMPITURA? (1-7)";C1
2110 IF C1<>VAL "0" AND C1<>VAL "1" AND C1<>VAL "2" AND C1<>VAL "3" AND C1<>VAL "4" AND C1<>VAL "5" AND C1<>VAL "6" AND C1<>VAL "7" THEN GO TO VAL "2100"
2115 IF C1="4" THEN GO TO 2400
2120 INPUT "DENSITA' CAMPITURA? (1-30)";L
2130 IF L<>VAL "30" THEN GO TO VAL "2120"
2140 FOR S=1 TO I-2 STEP 1: LET

```

- H Traccia la scala a sinistra.
- E Evidenzia valori. Ripetuto, cancella linee verticali.
- F Riquadra l'immagine (\*). Ripetuto, elimina il riquadro.
- R Solo Dia: traccia reticolo di riferimento sul diagramma. Ripetuto, lo cancella (\*).
- V Cambia la scala dei valori. Si possono tracciare dia e isto con scale di valori diverse nello stesso grafico.
- X Cancella i valori nella scala (utile se scale diverse).
- O Solo Dia e Isto corti: traccia scritte sul lato destro.
- P Solo Dia e Isto corti: cancella scritte sul lato destro.

### CAMPITURE ISTOGRAMMI

- 1 Righe verticali a densità variabile su tutte le barre.
  - 2 Righe orizzontali a densità variabile su tutte le barre.
  - 3 Tratteggio su tutte le barre.
  - 4 Barre singole: righe verticali.
  - 5 Barre singole: righe orizzontali.
  - 6 Barre singole: tratteggio.
- N.B. La campitura viene realizzata solo sull'ultimo dia o isto tracciato. Per i dia, nel caso di più dia in uno stesso grafico, partire con campiture non fitte.
- 8 Memorizza l'immagine video. La memorizzazione avviene automaticamente: a) dopo il



tracciamento di un dia o isto; b) se si richiamano le istruzioni; c) se si richiama la tabella valori.

- 9 Richiama l'immagine video.  
 Y Salva l'immagine su nastro.  
 S Stop: accede al basic. Per tornare all'immagine dare GOTO 8000 (per gli isto) o GOTO 9000 (per i dia) quindi premere il tasto 9.  
 K Copia l'immagine su carta:  
 (\*\*) 0 Copy normale,  
 1 Inverse copy,  
 2 Copy con retinatura corrispondente al colore.

N.B. I comandi funzionano sia in maiuscolo che in minuscolo.

(\*) "F" e "R" SHIFTATI confermano definitivamente RIQUADRO o RETICOLO.

N.B. Chi non usa la stampante Alphacom 32 o ZX printer

(\*\*) deve modificare le linee 9115 e 8145 nel modo seguente:  
 8145 IF I\$ = "R" OR I\$ = "K" THEN COPY: GOTO VAL "8000"  
 9115 IF C\$ = "R" OR C\$ = "K" THEN COPY: GOTO VAL "9000"

Se vogliamo fare un disegno e sapere in seguito se un punto è stato tracciato oppure no, possiamo risolvere il problema in due modi differenti. Il primo consiste nel memorizzare, per esempio in una matrice bidimensionale, le coordi-

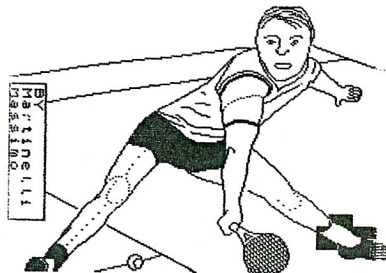
nate dei punti tracciati, ma questo implica un'enorme spreco di memoria altrimenti, seconda soluzione, utilizzare l'utilissima funzione Basic POINT. Per capire il funzionamento di questo comando osserviamo la linea 2060. La funzione POINT ha bisogno di due operandi: la coordinata X e Y del punto da testare. Il risultato di POINT è 1 se quel punto è stato tracciato cioè se il pixel ha il colore dell'inchiostro o 0 se il pixel ha il colore della carta, cioè se quel punto non è stato tracciato. Questa funzione è quindi molto utile per una completa e veloce gestione della memoria video. Purtroppo i programmatori non ne fanno un uso frequente ma ora che ne conoscete il suo funzionamento ricordatevi che esiste!



# TENNIS

(48K)

di M. Martinelli



**G**iocare a tennis non è poi così difficile, prova a batterti contro il tuo Spectrum! L'omino del computer si trova in basso allo schermo mentre il tuo è quello visualizzato nella parte alta del campo. Prima d'iniziare una partita dovrai immettere la difficoltà di gioco (1 = facile, 9 = difficile); quest'ultima si riferisce alla posizione che l'omino del computer può tenere in campo (da fondo a rete) e aumenta in relazione alla grandezza del numero immesso. Dopo che il campo di gioco sarà stato disegnato, basterà premere un tasto qualunque per poter cominciare la partita: la battuta inizialmente spetta al giocatore. Il colpo del tuo omino viene regolato dallo spostamento del giocatore stesso mediante i controlli da tastiera.

Tasti:  
 P = destra Q = su  
 O = sinistra A = giù

In questo gioco sono bastati solo 8,5 Kbyte di L/M per ottenere il gioco del tennis. Vediamo dove e come inizia il programma. Il programma Basic carica i blocchi di l/m e successivamente manda in esecuzione l'assembler con l'istruzione RANDOMIZE USR 43520. L'indirizzo decimale 43520 corrisponde in esadecimale ad AA00H. A questo indirizzo l'istruzione CALL A800H fa eseguire la su-

A80D 28FA JR Z,A809

Vediamo di descriverne il loro funzionamento. Il registro indice IY è utilizzato dall'interprete Basic ed il suo valore è sempre 23610, corrispondente alla variabile di sistema ERR-NR. L'istruzione BIT 5, (IY+01) testa il valore del quinto bit della locazione di memoria puntata da IY+1, cioè 23611. Questa locazione è la variabile di sistema FLAGS di cui

```

A800 1139A8
A803 01A200
A806 CD3C20
A809 FDCB016E
A80D 28FA

```

```

LD DE,A83A
LD BC,00A2
CALL 203C
BIT 5,(IY+01)
JR Z,A809

```

broutine all'indirizzo A800H. Qui notiamo che con due sole istruzioni è possibile svolgere l'equivalente Basic:  
 nn IF INKEY\$="" THEN GOTO nn

Le istruzioni interessate sono:  
 A809 FDCB016E BIT 5,(IY+01)

ogni bit ha un particolare significato. L'istruzione BIT setta il flag di zero Z se il bit testato è 0. Nel nostro caso se il bit 5 di (23611) è 0 vuol dire che non è stato premuto nessun tasto.



# PER ELISA

(16K)

di M. Piccinato



**A**ppena caricato il programma premi un qualsiasi tasto e magicamente dall'altoparlante del tuo Spectrum uscirà la famosa melodia di Ludwig van Beethoven: *Per Elisa*. Finita l'esecuzione per riascoltare ancora la musica digita RUN e poi ENTER. Buon ascolto!

Suonare con lo Spectrum non è facile, non solo perché ha limitatezze hardware ma anche software. Infatti l'unica istruzione sonora è BEEP t,n. Dove t è la durata ed n la nota. Questo programma pur essendo scritto in Basic riesce a riprodurre della musica molto bene, utilizzando esclusivamente l'istruzione BEEP. Ma vediamo come deve essere usata.

Il parametro t (durata) è una variabile o un'espressione numerica positiva compresa tra 0 e 10.4 ed esprime il numero di secondi della durata della nota. Il parametro n (nota) è un valore numerico compreso tra -60 e +69. La nota

```
15 9 BORDER 0: PAPER 0: INK 7: 0
15 : PRINT AT 10,9:"STAI ASCOLTANDO
15 : "PER ELISA" :
15 : "Ludwig van Beethove
15 FOR L=0 TO 1
15 BEEP .2,15: BEEP .2,15: BEE
15 : BEEP .2,15: BEEP .2,15: BEEP .2,15:
15 : BEEP .2,11: BEEP .2,14: BEEP .2
15 : BEEP .2,9
15 : BEEP .2,-15: BEEP .2,-8: BEE
15 : BEEP .2,0: BEEP .2,4: BEEP
```

0 corrisponde al DO dell'ottava centrale. Per passare da una ottava a quella superiore bisogna sommare 12 alla nota oppure sottrarre 12 se si vuole passare all'ottava inferiore.

Ecco i numeri dell'ottava centrale e il loro significato:

- 0 = DO
- 1 = DO# o REb
- 2 = RE
- 3 = RE# o MIb
- 4 = MI
- 5 = FA
- 6 = FA# o SOLb

- 7 = SOL
- 8 = SOL# o Lab
- 9 = LA
- 10 = LA# o Sib
- 11 = SI

Per tradurre una musica sullo Spectrum conviene sempre procurarsi uno spartito ed indicare sotto ad ogni nota il suo numero e a lato il valore numerico della durata. Tradotta la musica è sufficiente introdurre i valori in blocchi DATA e, tramite un ciclo FOR-NEXT, leggerli (READ) ed eseguirli con BEEP.



# PUNCHBALL (48K)

di F.F. Fantazzini

**M**isura la tua forza con il punchball dei Luna Park! Dopo aver inserito il tuo nome e scelto la categoria (pesi leggeri, medi o massimi) devi premere alternativamente i tasti «P» e «O» in modo da far salire l'indicatore

della FORZA che è presente nello schermo in basso a destra. Quando la FORZA ha raggiunto un buon livello, devi spostare l'indicatore della direzione usando il tasto «W» per la destra e il tasto «Q» per la sinistra. Quando l'indicato-

re della direzione è nella posizione centrale, devi premere il tasto «M» ed automaticamente partirà un pugno dalla destra dello schermo che andrà a colpire il pallone. A questo punto viene calcolato il punteggio parziale della partita

che dipende dalla potenza e dalla direzione del pugno. Dopo tre turni la partita finisce e ti verrà comunicato se il tuo punteggio è stato sufficiente per entrare in classifica. Via con il pugno!

Vi sono qui due variabili di sistema, REPDEL e REPPER, che controllano la ripetizione dei tasti. Modificando adeguatamente i loro valori si possono ottenere diversi effetti che possono anche proteggere il vostro software da occhi indiscreti!

Notiamo alla linea 9035 che anche l'autore di questo programma ha utilizzato queste variabili. REPDEL (locazione 23561) contiene il tempo di ritardo prima che un tasto premuto venga ripetuto. All'atto dell'accensione del sistema il contenuto di REPDEL è 35 ed indica un ritardo pari a 35/60 secondi. Come detto prima queste variabili possono essere modificate, provate per esempio POKE

23561,1 e la ripetizione avverrà immediatamente rendendo difficile il controllo del cursore. Inserendo invece il valore 0 (POKE 23561,0) si elimina il repeat dei tasti quindi per scrivere più volte una lettera bisognerà premere il suo tasto tante volte quante lettere vogliamo scrivere. Il massimo valore inseribile in REPDEL è 255 che provoca un ritardo di circa 5 secondi.

La variabile REPPER (locazione 23562) invece, contiene il ritardo tra una ripetizione e l'altra di un tasto quando questo è già premuto. Anche in questo caso una POKE 23562,0 elimina la ripetizione dei tasti mentre il valore 50 porta l'intervallo a 1 secondo. Per muoversi velocemente con il cursore in una istruzione molto lunga potete precedere l'editing della linea con una POKE 23562,1.

```

9000>INK 7: OVER 0: BORDER 0: P
9001>FOR 0: CLS
9005>POKE 23561,1: POKE 23562,1:
9006>LET v$="": FOR z=1 TO 32
9007>PRINT AT b(z),c(z);a$(z)
9008>NEXT z
9009>LET v$=176: LET xs=1: LET
9010>LET ps="Inserisci il tuo n
9011>INK 6: GO SUB 9700: INK 7:
9012>PRINT AT 0,0;"M=Corregge"
9013>LET ps="M=Fine"
9014>PRINT AT 0,22;"6=Sinistra":
9015>PRINT AT 1,22;"7=Destra";AT 2,22;"ENT
9016>LET ps="

```



# SLOT MACHINE (16K)

di I. Puddu

Spostiamoci a Montecarlo e stentiamo la fortuna alla macchinetta dei miracoli. Si puntano dei soldi e poi si preme ENTER. A questo punto, in base alla sola fortuna, potremo vincere o... perdere.

Chiamando A, B e C le tre figure che appaiono sullo schermo, le vincite sono così assegnate:

A=B o B=C 5 volte la puntata

A=C 2 volte la puntata

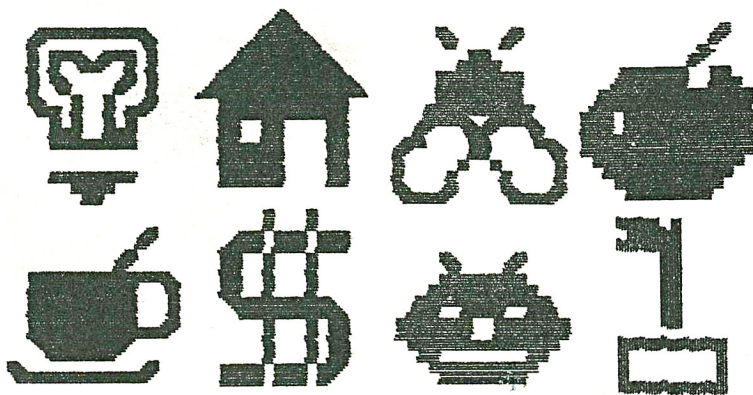
A=B=C 50 volte la puntata

\$\$\$ 100 volte la puntata.

Se invece A è diverso da B e B è diverso da C, si perde quanto si è puntato.

Buona fortuna!

Questo programma è interessante per due aspetti: è corto ed ha una buona e veloce grafica. Tralasciamo il primo punto e soffer-



```

140>FOR S=1 TO 5
150 LET A(S)=1+INT (RND*14)
155 RESTORE 3010: FOR I=1 TO a(
S): FOR f=0 TO 7: READ b$: PRINT
BRIGHT 1;AT 9+f,2+x;b$: NEXT f:
BEEP 1/100,15+2*I: NEXT I
190 FOR K=1 TO 20: BEEP .002,30
+x: NEXT K
195 LET X=X+10
200 NEXT S
210 LET vid=1: IF a(1)=a(2) AND
a(2)=a(3) THEN LET vid=50: GO 3
215 GO TO 140: POKE 23614,254: GO TO 2

```

**H**ai aperto una birreria e la vita è dura fin dall'inizio. Devi servire dei boccali di birra lanciandoli ai tuoi clienti. I clienti te li rimandano vuoti e tu devi prenderli al volo prima che cadano. Il gioco ha termine quando rompi 5 bicchieri oppure quando i clienti arrivano al banco dove tu servi.

Tasti:

1 = seleziona livello 1

2 = seleziona livello 2

6 = giù

7 = su

Come si vede per realizzare simpatici giochini non è necessario ricorrere per forza a lunghi listati o a routine in linguaggio macchina. Quello che conta maggiormente è l'idea e soprattutto la tecnica di programmazione.

Questo programma infatti, pur essendo molto breve è comunque divertente e, dal punto di vista tecnico, strutturato bene. Per esempio, guardiamo semplicemen-



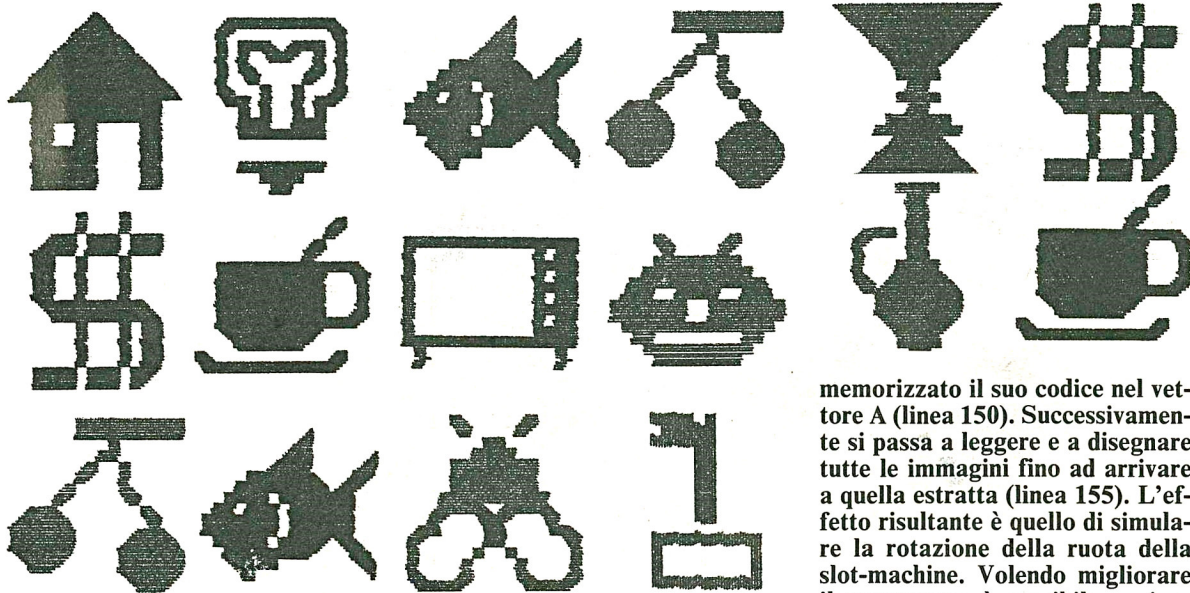
# BARMAN (16K)

di R. Vidoli

te la presentazione, è di effetto la scritta rotante a caratteri «BARMAN». Come l'avreste realizzata voi? Con 10, 15, 20 istruzioni? No, l'autore l'ha realizzata con soli 4 comandi basic: FOR-NEXT, POKE, PRINT e BEEP per la musichetta. Queste istruzioni sono alle linee 9041-9044. Vediamo di spiegare questo breve ma interessante pezzo di programma.

Innanzitutto dobbiamo dire che

l'insieme si basa sulla modifica del set dei caratteri. Nelle variabili di sistema, alla locazione 23606-7, è memorizzato l'indirizzo di base del set dei caratteri normali dello Spectrum, cioè quelli che abbiamo subito dopo l'accensione. Questo indirizzo è 15360 che è memorizzato come 0 in 23606 e 60 in 23607. Infatti  $60 * 256 = 15360$ .



miamoci alla descrizione della grafica.

La visualizzazione dei vari oggetti e frutti sfrutta una tecnica decisamente interessante. La parte di programma che gestisce la grafica si trova dalla linea 140 alla linea 200.

Prima di passare alla descrizione di queste linee dobbiamo dire dove e in che modo sono stati me-

morizzati i vari disegni. Essi sono memorizzati nelle istruzioni DATA a partire dalla linea 3010 tramite dei caratteri predefiniti UDG. Vediamo ora come ogni disegno viene prelevato e visualizzato.

Il ciclo FOR-NEXT alla linea 140 gestisce la visualizzazione di tre immagini. L'immagine da visualizzare viene estratta casualmente tra le 14 disponibili e viene

memorizzato il suo codice nel vettore A (linea 150). Successivamente si passa a leggere e a disegnare tutte le immagini fino ad arrivare a quella estratta (linea 155). L'effetto risultante è quello di simulare la rotazione della ruota della slot-machine. Volendo migliorare il programma è possibile aggiungere nuovi oggetti, e quindi nuove istruzioni DATA, in coda a quelli già esistenti rendendo il gioco ancor più difficile. Facendo questa modifica bisogna anche cambiare il numero di oggetti alla linea 150, che non sarà più 14 ma la nuova somma degli oggetti disponibili. Attenzione: il programma è protetto, per vederne il listato digitare all'input della puntata la scritta «list©» (© = copyright).



# EXERCISES OF ENGLISH GRAMMAR (48K)

di V. Vernon

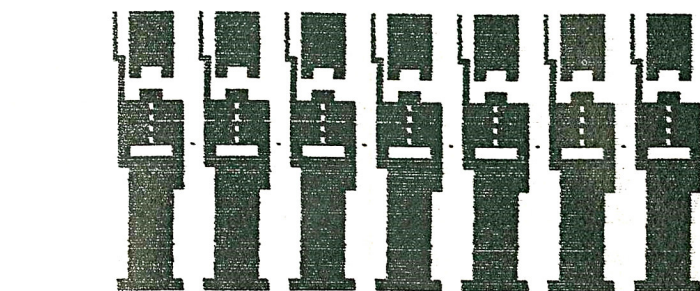
(2ª parte)

**E**cco il piatto forte della grammatica inglese: i verbi irregolari.

Vi sono 4 opzioni. Le prime due riguardano tutti i verbi: S per vederli tutti, a gruppi di 16 alla volta, completi di passato semplice e di participio passato nonché di traduzione in italiano. Così puoi studiare più agevolmente, tanto più che dopo si può passare all'opzione T (per training) per essere severamente interrogato e «punteggiato» in base alle risposte esatte ed alla prontezza di risposta. Le altre due opzioni sono di controllo su un verbo scelto da te: I per vedere i tempi passato e participio passato e la traduzione; J per coniugarlo nei tempi attivi. In caso di BREAK dare GOTO 4000.

La struttura del programma è chiara e lineare e comprende anche simpatiche animazioni grafiche, come quella nell'introduzione gestita dalla linea 4330 alla linea 4360. Non staremo a descrivere questa parte di programma quanto invece ci soffermeremo sulla descrizione di una particolare istruzione: PAUSE. Apparentemente inutile, questo comando si rivela invece indispensabile in diverse situazioni di programmazione. La sua struttura sintattica è la seguente: PAUSE n, dove n è una variabile o una espressione numerica compresa tra -5 e 65535. L'effetto del comando PAUSE è quello d'interrompere l'esecuzione del programma per un tempo pari a n/50 secondi.

Quindi con n = 50 l'interruzione sarà di 1 secondo mentre con n = 65535 l'interruzione sarà più di 21 minuti. Un caso particolare è quando n è uguale a 0, con questo valore l'interruzione è illimitata.



```
4300>PRINT " PURTROPPO L'UNICO
SISTEMA VALIDO A TUTT'OGGI ESSE
CITATO PER APPRENDERE TALI VERBI
E FORME RIMANE QUELLO DI STUDIAR
SELI A MEMO-RIA"
4310 PRINT " VE LA SENTITE ? A
E SI PREMETEUN TASTO, E VEDRETE
IL PIANO DISTUDIO CHE VI PROPOR
NIAMO. SE NO,DATE <BREAK> E <NEW
> E LASCIATEL'AULA"
4330 GO SUB VAL "10": CLS : LET
U1=VAL "50": LET U2=VAL "215": L
ET P1=2: LET P2=VAL "219": LET P
3=VAL "306": GO SUB VAL "610"
4340 PAUSE VAL "100": LET U1=VAL
"224": LET U2=VAL "216": LET P1
=VAL "202": LET P2=VAL "218": LE
T NM=NU: LET US=ZE: GO SUB VAL "
500"
4350 PAUSE VAL "50": LET U1=VAL
"224": LET U2=VAL "216": LET P1=
VAL "204": LET P2=VAL "218": LET
NM=VAL "9": LET US=ZE: GO SUB V
AL "600"
4360 PAUSE VAL "70": LET U1=VAL
"224": LET U2=VAL "216": LET P1=
VAL "222": LET P2=VAL "218": LET
NM=VAL "5": LET US=ZE: GO SUB V
AL "600"
```

L'effetto del comando PAUSE può essere interrotto, per continuare il programma, premendo un qualsiasi tasto. Appare chiaro come questa istruzione si riveli indispensabile in alcune situazioni, per esempio in un'attesa da parte del computer di un comando utente tramite la pressione di un tasto. La mancanza di questa istruzione porterebbe a sostituirla con le se-

guenti linee di programma:  
mm FOR J=0 TO n  
mm+1 IF INKEY\$<◇>"" THEN  
GOTO mm+3  
mm+2 NEXT J  
mm+3 resto del programma

Ma anche in questo modo l'istruzione PAUSE non sarebbe completamente simulata, infatti la condizione n=0 non è prevista dal programma sostitutivo.





# TOTOTOOLKIT (48K)

di G. Roncolato e R. Marroccu

**E**cco tredici routine in linguaggio macchina per analizzare e mettere a punto i propri programmi. Il programma può essere usato anche senza il programma Basic di accompagnamento, battendo NEW a caricamento avvenuto. In questo caso, per attivare le routine singolarmente, sarà necessario scrivere:

|                          |  |
|--------------------------|--|
| 1 Memoria libera         | PRINT USR 63000  |
| 2 Scambio caratteri      | POKE 63015, VECCHIO: POKE 63017, NUOVO: RANDOMIZE USR 63014  |
| 3 Elimina Rem            | RANDOMIZE USR 63057  |
| 4 Lista variabili        | CLEAR: RANDOMIZE USR 64207   |
| 5 Copia linee            | RANDOMIZE USR 63343 e poi rinumerare con la 13   |
| 6 Analizza programma     | RANDOMIZE USR 63762  |
| 7 Indirizzo linea        | PRINT USR 63891  |
| 8 Trasforma in maiuscole | RANDOMIZE USR 63904  |
| 9 Trasforma in minuscole | RANDOMIZE USR 63963  |
| 10 Confondi listati      | RANDOMIZE USR 64022  |
| 11 Riordina listati      | RANDOMIZE USR 64207  |
| 12 Accorcia programmi    | RANDOMIZE USR 64386  |
| 13 Rinumeratore          | POKE 64814, intervallo: POKE 64816, inizio-256*INT(inizio/256): POKE 64817,INT(INIZIO/256):RANDOMIZE USR 64528 |

```
:003 IF JYXH=VAL "3" OR JYXH=VAL
"17" THEN LET JYXH=VAL "4"
:004 IF CODE INKEY$=VAL "13" TH
N POKE VAL "23624",bord: POKE U
L "23693",col: STOP
:005 GO TO 10002
:006 RESTORE 9999: LET JYXH=JYX
-VAL "3": FOR K=VAL "1" TO JYXH
READ J: NEXT K: IF JYXH=VAL "2"
THEN GO TO 10015
:007 IF JYXH=VAL "3" THEN POKE J
AL "23624",bord: POKE VAL "23693
```

Vediamo ora a che cosa servono:  
 1: indica la memoria libera in bytes.  
 2: serve per sostituire in un programma tutti i caratteri di codice VECCHIO con quelli di codice NUOVO.  
 3: elimina tutte le REM del

programma.  
 4: stampa tutte le variabili del programma.  
 5: copia blocchi di linee in un altro punto del programma.  
 6: stampa il numero delle linee e delle istruzioni di un programma.  
 7: restituisce l'indirizzo della linea puntata dal cursore.  
 8: trasforma tutte le stringhe di un

programma in maiuscole.  
 9: trasforma tutte le stringhe di un programma in minuscole.  
 10: cambia i numeri del listato in altri a caso, lasciando inalterata la notazione a 5 bytes. Può essere un buon metodo di protezione.  
 11: riordina i listati secondo la no-

tazione a cinque bytes. È l'inverso di 10.

12: accorcia i programmi trasformando gli 0 in NOT PI ed i numeri in VAL «numero».

13: rinumeri i programmi. All'inizio potrà forse essere più facile utilizzare le routine con il programma di accompagnamento. In questo caso, dovrai muovere il cursore sulla linea desiderata (tasti 7 e 6) e premere il tasto 0.

Osservando il listato Basic notiamo delle linee con cifre strane, questo perché hanno un numero superiore a 9999. Sappiamo infatti che lo Spectrum, accetta numeri di linea compresi tra 1 e 9999. È possibile però oltrepassare questi limiti e quindi numerare una linea come 10000 (anche se poi si vedrà sullo schermo :000 e non 10000) ed eseguirla con GOTO 10000 o altre istruzioni.

Per numerare una linea con valore superiore a 9999 è sufficiente inserire nei due byte del numero di linea il valore appropriato. Supponiamo per esempio di voler inserire la linea 10000 come ultima istruzione di programma contenente la scritta di copyright. La procedura per inserire questa linea è molto semplice. Accendete il computer e come prima cosa digitate la seguente linea:

10 REM COPYRIGHT 1986.

A questo punto sostituiamo il numero 10 con 10000 con i seguenti due comandi diretti:

POKE 23755,39: POKE 23756,16

Ora la linea REM sarà preceduta dal numero 10000 visualizzato come :000. Alla base di tutto sta il fatto che i primi due bytes di una linea Basic sono il numero della stessa. Il primo byte contiene il valore più significativo.



# LOAD BANK

Hai programmi originali, esclusivamente pensati e fatti da te? Mandaceli in visione e, se verranno pubblicati, saranno certamente compensati a partire da lire 100 mila in su (a seconda del tipo di programma). Se sei fantasioso, bravo e veloce oltre che ordinato, puoi arrivare a guadagnare un bel gruzzoletto! Naturalmente il tuo nome apparirà stampato sulla rivista come valido collaboratore. Fruga nella tua fantasia e mandaci non solo arcade ma anche utility inedite, giochi di società particolari, interessanti routine in linguaggio macchina ed avventure con grafica (invia sempre anche la soluzione comando per comando in sequenza) allegando anche caratteristiche tecniche dei programmi, listati, stampate di screen etc. Se decidi di inviare un programma (indirizza a Load'n'Run, c.so Vitt. Emanuele 15, 20122 Milano) segui queste regole:

- 1) salva il programma su entrambi i lati di una cassetta;
- 2) usa 1 cassetta per ogni programma;
- 3) scrivi il tuo nome, quello del programma e quello della macchina sulla cassetta;
- 4) accludi un foglio dattiloscritto dove spiegherai a cosa serve il programma, come si usa, quali tasti usare, lo scopo. Non proteggete i programmi! Dobbiamo guardarli dentro. Non mettete il vostro indirizzo all'interno del programma (basta «by... nome e cognome»).

Causa diversità hardware tra le differenti serie di Spectrum, mandaci programmi che non usino il comando IN per la lettura della tastiera; sostituisilo con un'altra istruzione Basic (per esempio INKEYS) o con una routine in L/M.

Rispondiamo sempre a tutti e, soprattutto, manteniamo la parola data. Coraggio dunque, fatevi vivi!

**ATTENZIONE:** I programmi inviati debbono essere assolutamente inediti, a noi ceduti in esclusiva. L'editore si riserva di rifiutare programmi anche già accettati e di non compensare programmi già pubblicati qualora si verificasse che gli stessi programmi (anche quando con titoli diversi) siano stati ceduti o pubblicati da altri.

